**NODE**

**Clear-stergem outputul pe care l vedem (cls)**

**Window object-referinta a browser window**

****

**Cu process.argv luam input de la tastatura, din linia de comanda**

**Cu yargs(package) vs normal**

**Graphical user interface, text, application, chat or text message

Description automatically generatedText

Description automatically generated**

**Facem comenzi care daca rulam cu hep o sa ne apara Text

Description automatically generated**

**Text

Description automatically generatedprima e numele comenzii(nu merge modificat), a 2 a descrierea(merge modifica denumirea), a 3 a handler-ul(functia, ce sa faca cand e apelata comanda)**

**demandingOption->inseamna ca campul ala e required(default e pe false)-> adica daca nu vrem required nici nu trb sa l punem**

**type-> obligam ca acel camp sa fie de un anumit tip**

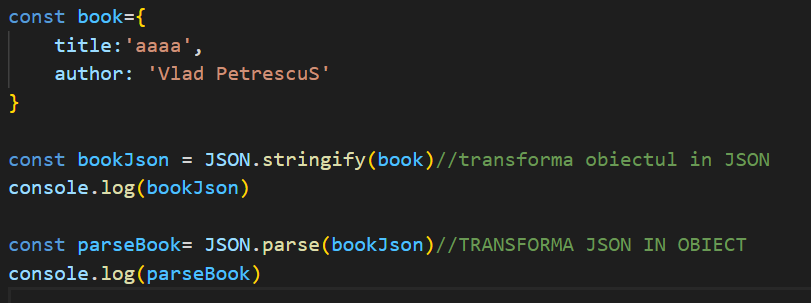
**Text

Description automatically generated**

**Text

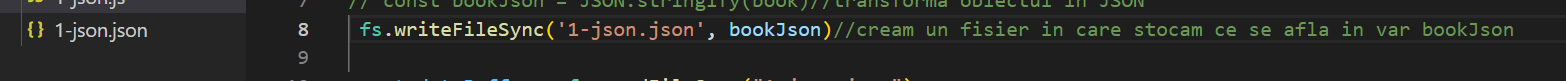
Description automatically generatedfara .parse nu putem afisa**

**JSON**

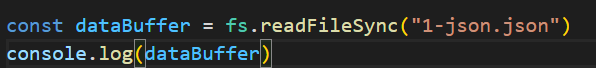
**Text

Description automatically generated**

**Putem accesa proprietatea doar pe obiect, nu si pe json (Adica nu merge bookJson.title, dar merge parseBook.title) !!!!!!!!!!!!!**

****

**Cu readFS->citim fisierul si am stocat in dataBuffer(Buffer inseamna biti si bytes).. trb sa folosim toString ca sa ne afiseze ce trb**

****

**Text

Description automatically generated**

**Add a note**

**Workfow: 1.Fct Load. 2fCT ADD 3.FCT SAVE

Description automatically generatedWorkfow: 1.Fct Load. 2fCT ADD 3.FCT SAVEWorkfow: 1.Fct Load. 2fCT ADD 3.FCT SAVE**

Arrow function (ES6)

Cu arrow function nu putem referential obiectul cu this.

Graphical user interface, text, application

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidence

**Debugg**-

Trb sa punem “debugger” in cod unde vrem

node inspect si continuam comanda normal

Intram pe chrome pe ://inspect

Text

Description automatically generated

Primele linii sunt cele mai importante.. in a 2 a ne zice exact unde e problema

**WEATHER APP (ASYNC)**

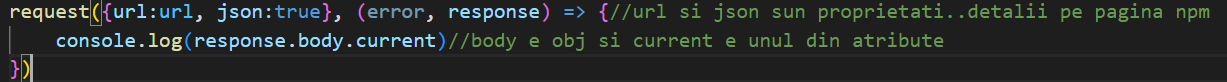
**Call stack->nodeApi-> callback queue**

**Call stack**

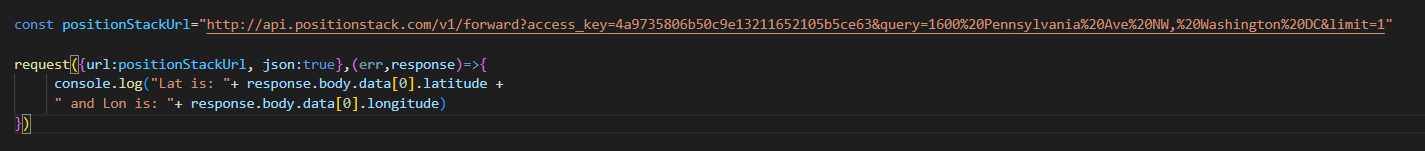
Intra pe sus si ca sa iasa trb sa iasa mai intai de sus in jos. Daca nu e vreun setTimeot sau cv sa astepte il afiseaza apoi iese din stack

Daca e-> intra in node api-> poate fie sa se intoarca in stack si sa faca logica sau se duce in queue, iar din queue in stack

**Http requests-**putem sa folosim un modul de node ca sa fie mai usor

****

Pt challenge folosim positionStack Api(free)



Acelasi sistem de lucru

**CallBack functions**

**Sunt functii pe care le punem ca parametru pt alte functii**

**Graphical user interface, text, application, email

Description automatically generated**

**Text

Description automatically generated**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Apelare:**

**Text

Description automatically generated**

**Callback Chaining-> Punem callback in callback**

**Text

Description automatically generated**

**OBJECT PROPERTY SHORTHAND**

**Normal**

**Text

Description automatically generated**

**SHORTHAND(practice daca au aceeasi denumire putem scrie doar o data)**

**Graphical user interface, text

Description automatically generated**

**DestructuringA picture containing graphical user interface

Description automatically generated**

**Putem redenumi noua variabilaGraphical user interface, application

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**In our app**

**For ex: destructing response from forecast**

**Aveam error, response ca param si peste tot unde am body, inainte aveam response.body…, acum doar body**

**Text

Description automatically generated**

**Fara ‘request’**

**Too much work .. easier with request library**